Journal of Network and Computer Applications 48 (2015) 71–86

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Energy-efficient algorithms for distributed storage system based on block storage structure reconfiguration

Bin Liao [a,b,*], Jiong Yu [b], Tao Zhang [c], Guo Binglei [b], Sun Hua [b], Changtian Ying [b]

[a] College of Statistics and Information, Xinjiang University of Finance and Economics, Urumqi 830012, PR China
[b] School of Software, Xinjiang University, Urumqi 830008, PR China
[c] Department of Medical Engineering and Technology, Xinjiang Medical University, Urumqi 830011, PR China

ARTICLE INFO

Article history:
Received 27 December 2013
Received in revised form
29 September 2014
Accepted 13 October 2014
Available online 31 October 2014

Keywords:
Cloud computing
Distributed storage system
Green computing
QoS constraints

ABSTRACT

As the underlying core infrastructure for cloud computing, distributed storage systems like the Hadoop Distributed File System (HDFS) are the foundation of all kinds of cloud services. However, designers of the ever-expanding systems have ignored the problem of high energy consumption, causing serious environmental and economic problems. The data availability and performance Quality of Service (QoS) requirements make it hard to use existing energy-saving technologies to solve the problem. After researching the data block's storage structure and mechanism, and the relationship between the server's status and the data block's availability, the method to solve the problem of ensuring data availability and performance QoS requirements is proposed. The energy-saving model for the distributed storage system is defined. The algorithm divides the RACK into two distinct storage areas, Active-Zone and Sleep-Zone, reconfiguring the data storage structure using the block storage structure reconfiguration algorithm. To save energy, we turn the servers in Sleep-Zone to sleep mode while the workload is low. Numerical analysis and experimental results demonstrate that the energy-efficient algorithms improved the energy efficiency for the distributed storage system.

1. Introduction

1.1. Background and motivation

Cloud computing, the long-held dream of "computing as a utility", has opened up the new era of future computing, has transformed a large part of the information technology (IT) industry, and has reshaped the purchase and use of IT software and hardware (Foster et al., 2008; Buyya et al., 2009; Armbrust et al., 2010; Mell and Grance, 2010; Subashini and Kavitha, 2011). As an emerging model for distributed utility computing, cloud computing has become commercially attractive and its use is growing since it promises to reduce maintenance and management costs for its customers in comparison with traditional data centers. Clouds are normally composed of large and power-consuming data centers designed to support the elasticity and scalability required by customers. A substantial percentage of these data centers run large-scale data-intensive applications, and systems such as MapReduce (Dean and Ghemawat, 2008), based on such Google File System (GFS) (Ghemawat et al., 2003), and HDFS (Borthaku, 2007) are distributed storage systems that have emerged as important infrastructures for building cloud services or applications.

However, with dramatically increasing demand on computing and storage systems, IT infrastructures have been scaled up tremendously, which results in a huge amount of energy consumption, heat dissemination, greenhouse emissions, and even a contribution to climate change. A report (Brown, 2007) by US Environmental Protection Agency (EPA) points out that the energy consumed by data centers has doubled in the period between 2000 and 2006 and estimates another two-fold increase from 2007 to 2011 if the servers are not used in an improved energy-efficient scenario. Moreover, distributed storage systems such GFS and HDFS, are all designed with mechanisms to ensure reliability, load balancing, fault tolerance, etc., but they ignore energy consumption factors that can have an impact on energy efficiency, as even idle machines remain powered on to ensure data availability. Google's server utilization and energy consumption study (Barroso and Hölzle, 2007) reports that energy efficiency peaks at full utilization and significantly drops as the utilization level decreases. Hence, power consumption at zero utilization is still considerably high (around 50%). Essentially, even an idle server consumes about half its maximum power, a fact that justifies the technique of switching idle servers to sleep mode to reduce the total power consumption. The main reasons causing the low

* Corresponding author at: College of Statistics and Information, Xinjiang University of Finance and Economics, Urumqi 830012, PR China.
E-mail addresses: liaobin665@163.com (B. Liao), yujiong@xju.edu.cn (J. Yu), 13999894800@126.com (T. Zhang), yingchangtian@xju.edu.cn (C. Ying).

http://dx.doi.org/10.1016/j.jnca.2014.10.008
1084-8045/

energy utilization in Google servers is that the GFS's load balancing algorithm averages its distribution of tasks to all servers and uses the replica mechanism to ensure data availability, which improves the usability and availability of the system but does not take the relationship between resource utilization and energy efficiency into account. Energy-aware algorithms should distribute the tasks to the appropriate number of servers, and unused servers should be able to turn to sleep mode without affecting data availability and the performance of QoS requirements while keeping resource utilization low. Thus, dynamically reconfiguring the block storage structure to ensure proper performance of the QoS requirements of data while turning the idle servers to sleep mode is the best way to improve the energy-efficient utilization of a data center which with a distributed storage system, and is an important element of the infrastructure for building cloud services.

1.2. Contributions

In order to achieve energy savings in distributed storage system environments, there are three important problems that must be solved. (1) A solution must be found to achieve energy savings using the technique of switching idle servers to sleep mode without decreasing the performance of QoS requirements for data files or tasks. (2) A solution must be found to determine which DataNodes should be switched to sleep mode to save energy consumption and when the switch should take place. (3) A solution must be found to control the data transfer cost for the block storage structure reconfiguration process.

In this paper, we address the issue of power and energy conservation for clusters or data centers that use a distributed storage system as an important infrastructure for building cloud services or applications. Although we focus on HDFS (Hadoop Distributed File System) especially, our approach can also be applied to other similar systems such as GFS, Luster, MooseFs, CarrierFs, and so on. Our contributions can be summarized as follows.

(1) By studying the cluster structure, data block storage mechanism, the relationship between DataNodes' status, and data blocks' status and availability, etc., we defined the Cluster DataNode Matrix, DataNode State Matrix, File Block Matrix, Block Storage Matrix, Block State Matrix, and Block Availability Check Matrix, which modeled the metadata for energy-efficient algorithms.
(2) Traditional energy-saving technology is challenged by data QoS constraints in distributed storage systems; our proposed energy-efficient algorithms not only achieved energy saving, but negotiated the QoS constraints as well.
(3) We defined an energy-saving model for distributed storage system, that divides the RACK into two separate storage areas, Active-Zone and Sleep-Zone, reconfigures the storage structure by calculating each data file's active factor, and turns the DataNode servers in Sleep-Zone to sleep status in order to achieve energy savings.
(4) Numerical analysis and experimental results verified the efficacy of our algorithm while we created the energy consumption model for the distributed storage system.

1.3. Paper organization

The remainder of this paper is organized as follows. Related works are surveyed in Section 2. We give some related definitions in Section 3. Next, we present our energy efficient algorithms for distributed storage systems based on block storage structure reconfiguration in Section 4. Section 5 describes the power and energy models for distributed storage systems. Section 6 presents the numerical analysis and experimental results of our algorithms. Finally, in Section 7 we conclude by summarizing our contributions and touching on future topics in this area.

2. Related work

Traditional IT systems have been focused on performance, availability, and scalability. However, continuously increasing energy consumption, heat dissemination, and greenhouse emissions of IT systems have started to limit further performance growth. Therefore, the goal of the IT system design has been shifted to power and energy efficiency. Existing work mostly focused on hardware (Benini et al., 2000; Albers, 2010; Srivastava et al., 1996; Hwang and Wu, 2000; Wierman et al., 2009; Andrew et al., 2010; Lorch and Smith, 2001; Zeng et al., 2005), operating systems (Neugebauer and McAuley, 2001; Vardhan et al., 2009; Flinn and Satyanarayanan, 2004; Meisner et al., 2009), virtualization (Liao et al., 2012; Jang et al., 2011; Wang et al., 2011; Dasgupta et al., 2011), and data center (Srikantaiah et al., 2008; Garg et al., 2011; Stillwell et al., 2009; Song et al., 2009; Cardosa et al., 2009; Gmach et al., 2009; Buyya et al., 2010; Kim et al., 2009) levels to improve the energy efficiency of IT systems. In this study, we focus on the energy efficiency of data centers with a distributed storage system as an important infrastructure for cloud services or other applications. With the availability and performance QoS requirements being ensured, our algorithm divides the RACK into two distinct storage areas, the Active-Zone and the Sleep-Zone, reconfiguring the data storage structure using a block storage structure reconfiguration algorithm, and energy is saved when the servers in the Sleep-Zone are turned to sleep mode.

Current energy-saving technologies for distributed storage systems are generally focused on the hardware and software. Hardware energy-saving technologies (Vasudevan et al., 2009; Kim et al., 2010; Szalay et al., 2010; Lim et al., 2008; Yin et al., 2010; Harnik et al., 2009; Greenan et al., 2008) reduce the total energy consumption of the system by using and energy-efficient hardware device (e.g., SSD, ARM architecture based CPU); and software energy-saving technologies (Leverich and Kozyrakis, 2010; Narayanan et al., 2008; Storer et al., 2008; Vasić et al., 2009; Chen et al., 2010; Chen et al., 2009; Oppenheim, 2010; Thereska et al., 2009; Maheshwari et al., 2012; Le et al., 2011; Wirtz and Ge, 2011) generally improve the energy efficiency of systems by effective resources scheduling.

Most recent works on energy conservation for distributed storage systems focused on exploiting low utilization periods and attempting to temporarily switch off some of the DataNodes to reduce the cluster's energy consumption. Those approaches improve the cluster's energy efficiency by dynamically adjusting the active server set according to the current workload. GreenHDFS (Kaushik and Bhandarkar, 2010) separated cluster servers into cold and hot zones and placed data in these zones according to data classification. GreenHDFS conserved energy by transitioning the servers in the cold zone to high energy-saving states. Leverich and Kozyrakis (2010) and Chen et al. (2010) studied energy efficiency with varying numbers of worker nodes and found energy-saving potentials for MapReduce applications based on HDFS. To achieve energy reduction, Maheshwari et al. (2012) dynamically reconfigured the cluster based on the current workload and turned cluster nodes on or off when the average cluster utilization rose above or fell below administrator-specified thresholds. Le et al. (2011) evaluated the energy efficiency of HDFS by power proportional data placement methods.